Introduction

The following set of exercises is designed to evaluate the development ability of sub-contractors. Please tackle this work as you you would commercial work from ourselves. We will consider the work that you provide to us here to be representative of what we can expect for anything that we do contract to you in the future. We do have a policy of refusing code that is not developed to a high enough standard by sub-contractors and we will expect future work to be of the same standard as the work that you deliver to us here, so please do not provide us exemplary code delivered to a standard that you are not able to commit to hitting for the future.

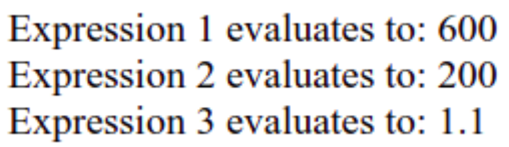
You should have received a folder, accompanying this document, including source code to go with the different exercises that are listed.

The work should be delivered back to us as:

- A zip file containing the code developed for each exercise.
- A short document outlining:
  - The list of participants who contributed to the test along with a description of their contribution (the roles that they hold within your structure are important to us. Their names are not)
  - The time that was required for each exercise
  - Any design notes outlining approaches taken to the different exercises with any relavent reasoning and any other notes that could help us to test and evaluate your solutions.

Exercise #1

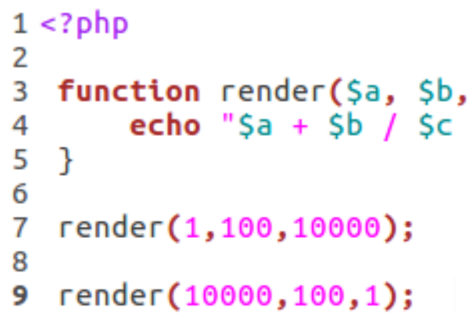
Implement the 'evaluate()' function to calculate the result of the expressions in the sample provided in file exercice-1.php. The output should look like this :



```
Expression 1 evaluates to: 600
Expression 2 evaluates to: 200
Expression 3 evaluates to: 1.1
```

Exercise #2

Here is an example program:



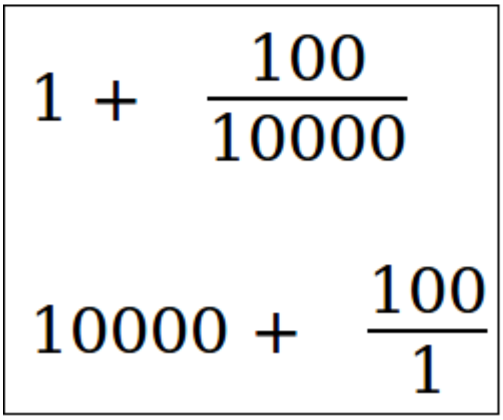
```php
<?php

 function render($a, $b, $c){
     echo "$a + $b / $c <br>";
 }

 render(1,100,10000);

 render(10000,100,1);
```

Using HTML and CSS, please update the above program to output something like this:



$$1 + \frac{100}{10000}$$

$$10000 + \frac{100}{1}$$

Exercise #3

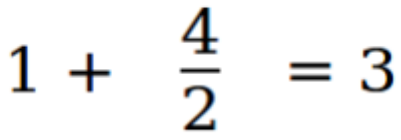
Recent browsers provide support for a CSS feature called FLEX containers.
If you are unfamiliar with them then the following references might be of interest to you.

- https://css-tricks.com/snippets/css/a-guide-to-flexbox/
- https://www.alsacreations.com/tuto/lire/1493-css3-flexbox-layout-module.html

Rework the solution to exercise 1 using FLEX containers.

Exercise #4

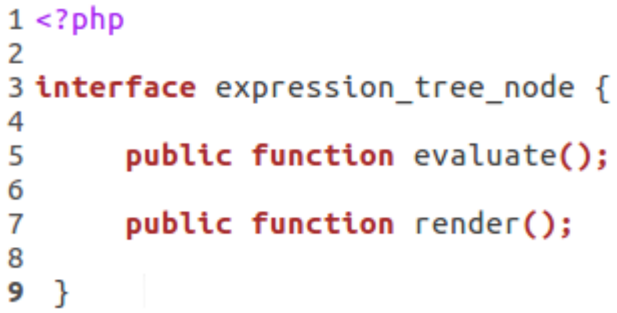
Combine your responses to exercises 1 and 3 to render arbitrary expressions and their solutions using HTML/CSS something like this:



$$1 + \frac{4}{2} = 3$$

Exercise #5

Rewrite the exercise 4 solution using a set of classes implementing the following interface:



```php
<?php

interface expression_tree_node {

    public function evaluate();

    public function render();

 }
```